Logistic regression

$\hat{y}^{(i)} = \sigma(w^T x^{(i)} + b)$, where

$$\sigma(z^{(i)}) = \frac{1}{1 + e^{-z^{(i)}}}$$

$z^{(i)} = w^T x^{(i)} + b$

$x^{(i)}$, $y^{(i)}$, $z^{(i)}$ } ith example

Given $\{(x^{(1)}, y^{(1)}), \ldots, (x^{(m)}, y^{(m)})\}$,

want $\hat{y}^{(i)} \approx y^{(i)}$

Loss error functn

$$L(\hat{y}, y) = -\left(y \log \hat{y} + (1-y) \log(1-\hat{y})\right)$$

* if $y = 1$: ~~$L(\hat{y}, y)$~~

$L(\hat{y}, y) = -(\log \hat{y})$

In order to get $L(\hat{y}, y)$ as small as possible, $\log \hat{y}$ must be large.

But $\hat{y}$ has limits $[0, 1]$

So $\hat{y}$ must be as close to 1 as possible. $\hat{y} \approx 1$.

* if $y = 0$

$L(\hat{y}, y) = -\log(1 - \hat{y})$

~~$-\log 1$~~

for $L(\hat{y}, y)$ to be small, $\log(1 - \hat{y})$ must

Cost funct^n / On entire training eg)

$$J(w,b) = \frac{1}{m} \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)})$$

$n \rightarrow$ no. of features

$m \rightarrow$ no. of training examples

$$\therefore J(w,b) = -\frac{1}{m} \sum_{i=1}^{m} \left[y^{(i)} \log \hat{y}^{(i)} + (1-y^{(i)}) \log (1-\hat{y}^{(i)}) \right]$$

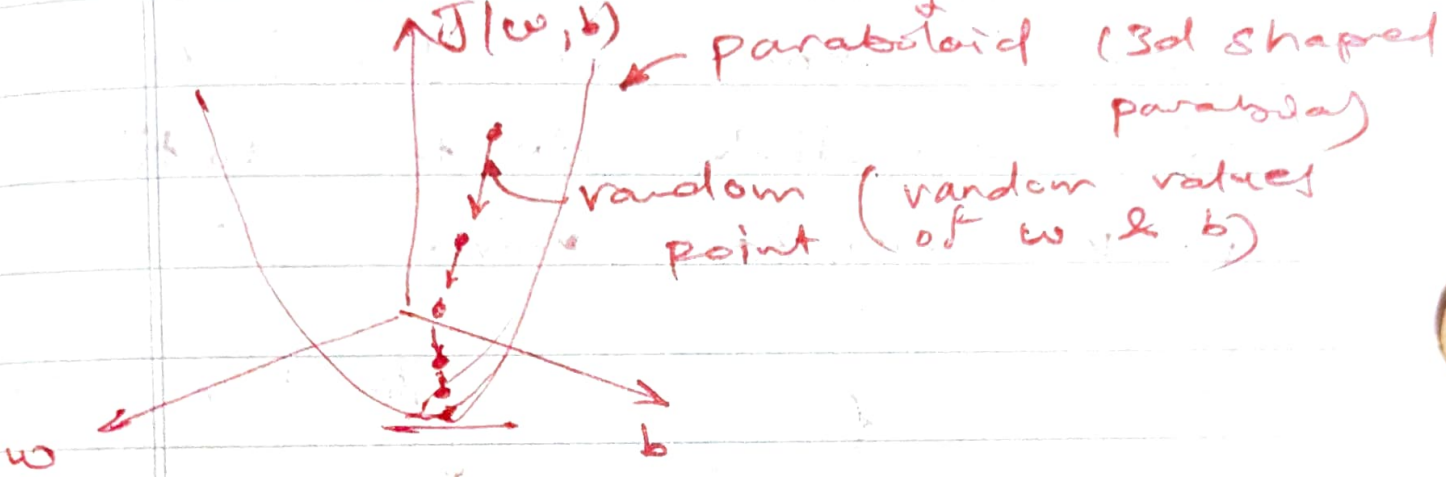
Gradient diteent :

what's loss function :- Calculating the loss for one training example

what's cost function :- Calculating the avg loss [i.e avg of loss funct^n] for one value of w & b. (const. value of w & b).

We need to minimize the ~~loss~~ cost function in order to get near perfect output. How can we do this?

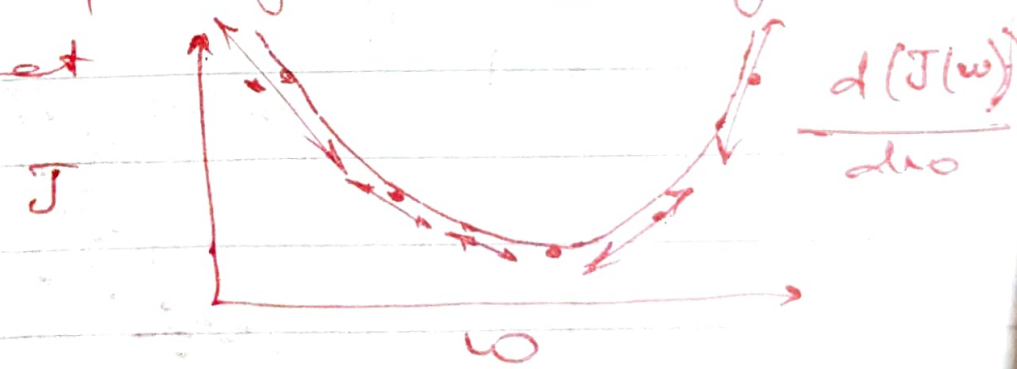
Imagine w and b are 1 dimentional. (though w can be multidimentional, since, training dataset has many features)



We want to reach the minimum of J(w,b) [to get min. loss].

How can we do this? We can do this by Gradient Descent

If we plot graph neglecting b, we'll get



Teacher's Signature

We'll calculate the slope at that random point & we'll nudge in the ~~min~~ direction of graph where we'll get min. J.
This is known as Gradient Descent.
We'll repeatidly do this until we've reached min. of J.

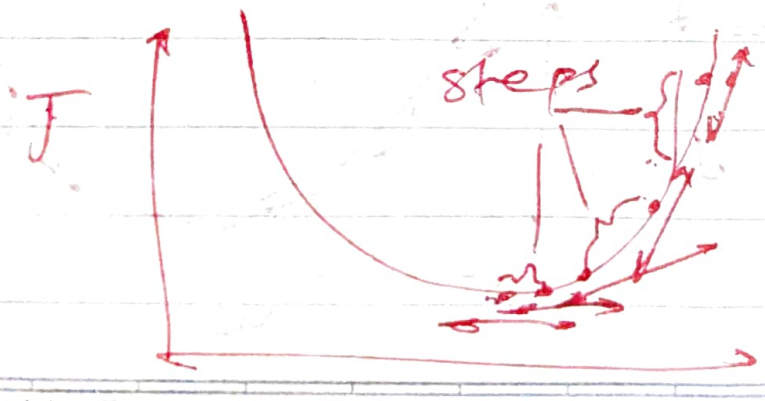
Repeat {

$$w := w - \alpha \frac{dJ(w)}{dw}$$

learning rate

} denotes

α ~~determines~~ the step which we'll have to take to calculate slope at that point.



Teacher's Signature

$w :=$ ____ ":=" update sign

↳ update w

No we'll do this for b as well

Repeat {

$$w := w - \alpha \frac{\delta J(w,b)}{\delta w}$$

$$b := b - \alpha \frac{\delta J(w,b)}{\delta b}$$

}

Since, J depends on multiple variables, we'll use partial derivative rather than normal derivative

PAGE NO.:
DATE: / /

$$a = \sigma(z)$$

$$\frac{da}{dz} = \frac{1}{1+e^{-z}}$$

$$= (1+e^{-z})^{-1}$$

$$= -1 \cdot (1+e^{-z})^{-2} \cdot [0 + -e^{-z}]$$

$$= \frac{e^{-z}}{(1+e^{-z})^2}$$

$$= \frac{e^{-z}}{1+2e^{-z}+e^{-2z}}$$

$$= \frac{1}{e^{z}+e^{-z}}$$

$$1-a = 1 - \frac{1}{1+e^{-z}}$$

$$= \frac{1+e^{-z}-1}{1+e^{-z}}$$

$$= \frac{e^{-z}}{1+e^{-z}}$$

Teacher's Signature

$$da = \frac{\delta L(a,y)}{\delta a}$$

$$= \frac{-y}{a} + \frac{1-y}{1-a} \quad \longleftarrow (1)$$

$$dz = \frac{\delta L(a,y)}{\delta z}$$

Using chain rule, we can write

$$= \frac{\delta L}{\delta a} \times \frac{\delta a}{\delta z} \quad \longleftarrow (2)$$

$$\frac{\delta a}{\delta z} = \frac{da}{dz} = \frac{d}{dz}(\sigma(z))$$

$$\sigma(z) = \frac{1}{1+e^{-z}} = (1+e^{-z})^{-1}$$

$$\frac{da}{dz} = -1(1+e^{-z})^{-2}[0 + (-1)e^{-z}]$$

$$= \frac{e^{-z}}{(1+e^{-z})^2}$$

We know,

$$a = \frac{1}{1+e^{-z}}$$

$$1-a = 1 - \frac{1}{1+e^{-z}}$$

$$= \frac{1+e^{-z}-1}{1+e^{-z}}$$

$$= \frac{e^{-z}}{1+e^{-z}}$$

$$a(1-a) = \frac{e^{-z}}{(1+e^{-z})^2}$$

$$\therefore \frac{da}{dz} = a(1-a) \quad \text{— (3)}$$

From (1), (2) & (3)

$$dz = \left(\frac{-y}{a} + \frac{1-y}{1-a}\right) \cdot a(1-a)$$

$$= -y(1-a) + (1-y) \cdot a$$

PAGE NO.:

DATE: / /

$$= -y + ay + a - ay$$

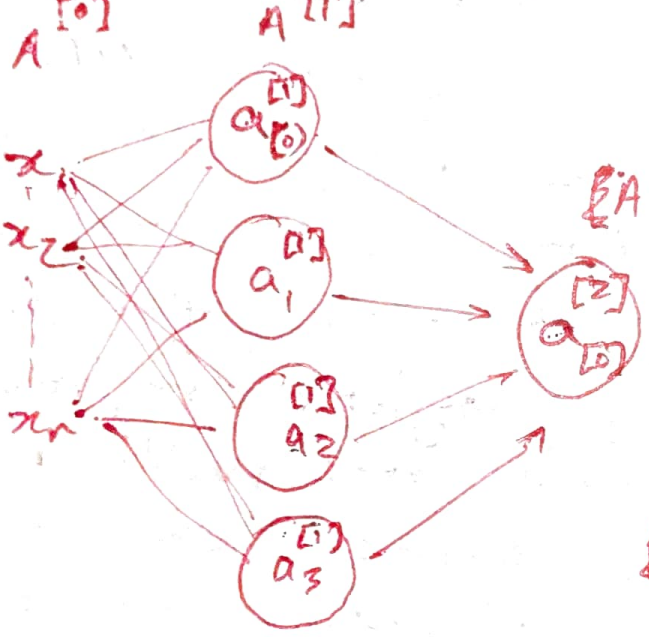
$$= a - y$$

$$\therefore \quad dz = a - y$$

Vectorizing across multiple examples

no. of test cases / examples = m

We know we have to perform



[i] → layer number or activation function layer number

(j) → activation function number in that layer

$$z = W^T x + b$$
$$a = \sigma(z)$$

→ One layer will calc. this & pass "a" to the next layer

Here x is also input from previous layer, & input layer.

for, we have to perform

$$z^{[1]} = W^{[1]} A^{[0]} + b^{[1]}$$

$$A^{[1]} = \sigma(z^{[1]})$$

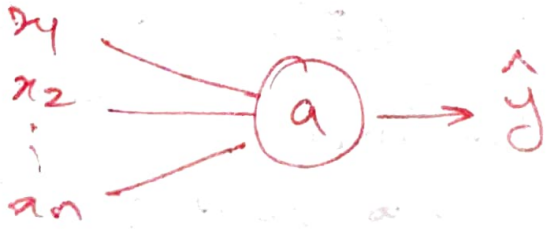
$$z^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$$

$$A^{[2]} = \sigma(z^{[2]})$$

The $A^{[2]}$ will result in final output & finally, it's a 2-layer NN.

We can iterate for each example/ training image/test case as it'll caute a lot of time to train the model.

As in previous logistic regression, we have vectorization, ~~not~~ we'll perform same here. In previous example though we had only 1 layer and only 1 activation function & no hidden layer



But here, we have an extra layer aka hidden layer & it has more than 1 activation functions. How can we do vectorization of the NN.

Say we have m training examples & each example has n_x features.

So $A^{[0]}$ or X matrix should be a matrix of (n_x, m). We can do this by stacking m examples column wise

In first NN, we have similar matrix X but here we'll call it as $A^{[0]}$ as explained before. We'll multiplying matrix $W^{[1]}$ with it, but $W^{[1]}$ has 4 activation functions. Therefore, dimensions of W must be $(4, n_x)$ as

$$W^{[1]}_{4 \times n_x} \times A^{[0]}_{n_x \times m}$$

no. of columns of $W^{[1]}$ and no. of rows of $A^{[0]}$ should match. We can achieve the dimension of $W^{[1]}$ by stacking up n_x rows

Teacher's Signature

4 times row with

$$W^{[1]} = \begin{bmatrix} — w_{(1)} — \\ — w_{(2)} — \\ — w_{(3)} — \\ — w_{(4)} — \end{bmatrix}_{4 \times n_x}$$

$$A^{[0]} = \begin{bmatrix} | & | & & | \\ x^{1} & x^{2} & \cdots & x^{m} \\ | & | & & | \end{bmatrix}_{n_x \times m}$$

training examples

and we can add b to it, we don't need to specify its dimensions as python we'll perform broadcasting for it.

$$A^{[1]} = \sigma(z^{[1]}) = W^{[1]} \cdot [A^{[0]}] + b^{[1]}$$

After calculating $z^{[1]}$, we can calculate $A^{[1]}$. Its dimensions would

Teacher's Signature

be $4 \times m$ (as $W^{[1]}_{4 \times n_x} \times A^{[0]}_{n_x \times m}$)

Now, we need to calculate $A^{[2]}$

$A^{[2]} = g(z^{[2]}) = W^{[2]} A^{[1]} + b^{[2]}$

What should be the dimensions for $W^{[2]}$.

$A^{[1]}$ has $4 \times m$ dimensions.

So $W^{[2]}$ must have 4 columns.

What about rows?

As $A^{[2]}$ is directly responsible for the output i.e. $\hat{y}$, it must be an unidimensional vector. $A^{[2]}_{1 \times 1}$

To get that, $W^{[2]}$ must have only 1 row.

∴ Dimensions of $W^{[2]} = (1 \times 4)$

$b^{[2]}$ will be a real number.

~~$\therefore A^{[2]} = W^{[2]} A^{[1]} + b^{[2]}$~~

$\therefore A^{[2]} = g(z^{[2]}) = W^{[2]} A^{[1]} + b^{[2]}$

Neural Network representation learning

$$z^{[1]}_{4\times 1} = W^{[1]}_{(4,3)} x_{(3,1)} + b^{[1]}_{(4,1)}$$

(or $a^{[0]}$)

$$a^{[1]}_{(4,1)} = \sigma(z^{[1]}_{(4,1)})$$

$$z^{[2]}_{(1,1)} = W^{[2]}_{(1,4)} a^{[1]}_{(4,1)} + b^{[2]}_{(1,1)}$$

$$a^{[2]}_{(1,1)} = \sigma(z^{[2]}_{(1,1)})$$

This is for 1 training example

Teacher's Signature

Note : $A \rightarrow$ for m examples
$a \rightarrow$ for 1 example

Forward propagation for multilayer NN (aka deep networks)

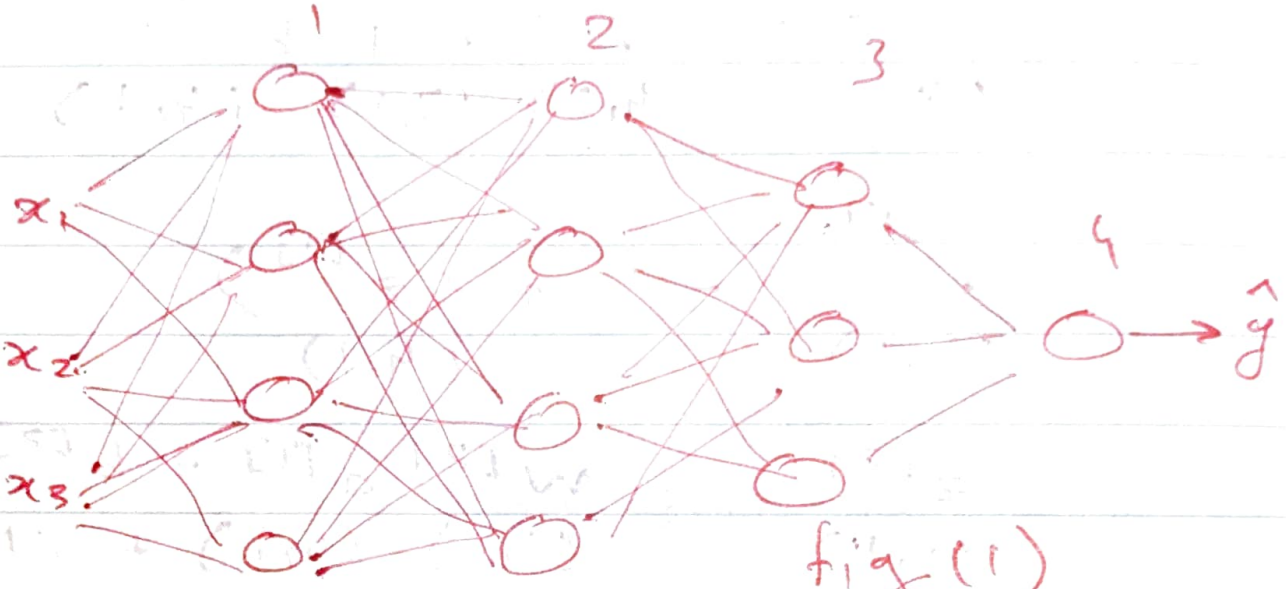


fig (1)

Forward propagation :

$$z^{[1]} = W^{[1]} A^{[0]} + b^{[1]}$$
$$A^{[1]} = g(z^{[1]})$$
$$z^{[2]} = W^{[1]} A^{[1]} + b^{[1]}$$
$$\vdots$$
$$z^{[4]} = W^{[4]} A^{[3]} + b^{[4]}$$
$$A^{[4]} = \hat{y} = \sigma(z^{[4]})$$

Teacher's Signature

General representation

$$z^{[l]} = w^{[l]} A^{[l-1]} + b^{[l]}$$

$$a^{[l]} = g^{[l]}(z^{[l]})$$

Vectorize it over l layers.

for $l = 1 ; l <= 5 ; l++$:

$$z^{[l]} = w^{[l]} A^{[l-1]} + b^{[l]}$$

$$A^{[l]} = g^{[l]}(z^{[l]})$$

Dimensions of z, w, b, dw, db throughout NN for diff. layers.

for fig (1), we have 3 input features

$$z^{[1]} = w^{[1]} \cdot x + b^{[1]}$$

x has dimensions $(3,1)$ — for 1 training example

$z^{[1]}$ has dim. $= (4,1)$ — since it has 4 nodes

$$\underset{(4,1)}{z^{[1]}} = \underset{(4,3)}{w^{[1]}} \cdot \underset{(3,1)}{x} + \underset{(4,1)}{b^{[1]}}$$

$w^{[1]}$ must have dim. $(4,3)$

We can say, $W^{[1]}$ has dimensions $(n^{[1]}, n^{[0]})$. $n^{[l]}$ → no. of nodes in layer l.

More generally, we can say,

$$W^{[l]} = (n^{[l]}, n^{[l-1]})$$

$$z^{[l]} = (n^{[l]}, 1)$$

$$b^{[l]} = (n^{[l]}, 1)$$

$$a^{[l]} = g^{[l]}(z^{[l]})$$

∴ dim. of $A^{[l]} = (n^{[l]}, 1)$

} for 1 training example

dim. of dw & db should be the same as W & b resp.